

Screencast: [26-virt-overview.webm](#) or [26-virt-overview.mp4](#)

LaUSAH REFERENCE - Chapter 24, Virtualization - Chapter 25, Containers

The goal of this lecture is to introduce you to the various virtualization products that are available for Linux and to compare and contrast them.

Related but distinctly different topics include: Terminal Services and Application Virtualization

Why Virtualize?

- Increase hardware utilization
- Improved resource management (configuration changes vs. hardware)
- Cost and energy savings
- Legacy OS / Applications won't run on new hardware
- Easier migration because hardware is abstracted
- Development and Testing
- Less painful upgrades with easy rollback
- Improved reliability with high availability / live or offline migration
- Security by isolating services

Use Cases

- Desktop user - Trying out new OSes, Linux distro hopping
- Small business - Servers and desktops
- Enterprise business - Datacenters for cost and energy savings
- Education - You have VMs and containers, right?
- Research - Easily make environments and simulations
- Cloud computing is heavily based on virtualization

Brief History of Virtualization

On mainframe computers IBM has had virtualization features built into their hardware since the 1960s.

In micro and personal computers the first virtualization product I heard about was a card for the Apple II that allowed running some DOS applications.

- Later Atari ST users could emulate Atari 8-bit computers.
- PC emulation was possible with PC-Ditto.
- Mac emulation was possible with Spectre.

Video game machine emulators are quite common... think MAME.

VMware released its first product in 1999.

Types of Hypervisors

A term you will see tossed around frequently when referring to both Full virtualization and paravirtualization is [hypervisor](#). The two distinct categories of hypervisors are:

1. Level 1 - *bare metal*
2. Level 2 - *hosted*

Many virtualization vendors offer a layered approach to their product line and may offer both type 1 and type 2 based products.

Products

[VMware](#) [1999] ([wikipedia](#))

Full virtualization

Type 2 - Windows, Mac, and Linux

VMware Player (no cost)

Server (no cost)

Workstation (cost)

Fusion (cost)

Type 1

ESX / Infrastructure (cost)

ESXi (no cost)

[SWsoft Virtuozzo](#) [2001] ([wikipedia](#)) - Later Parallels

OS virtualization

Linux version 2001 (cost)

Windows version 2005 (cost)

[Linux-VServer](#) [2001] ([wikipedia](#))

OS virtualization

Linux only (free software)

[Xen](#) / [Citrix](#) [2003] ([wikipedia](#))

Paravirtualization

Linux (free software, no cost, and cost)

Windows (maybe)

[OpenVZ](#) [2005] ([wikipedia](#))

OS virtualization (upstream of Virtuozzo)

Linux only (free software)

[Parallels](#) [2005] ([wikipedia](#))

Full virtualization

Type 2 for Mac, Linux & Windows

May have a Type 1?!?

OS virtualization (see Virtuozzo)

[VirtualBox](#) [2007] ([wikipedia](#))

Full virtualization

Type 2 for Mac, Linux, Windows, and Solaris

[KVM](#) [2007] ([wikipedia](#))

Full virtualization

Type 1.5 / Hybrid? Requires virt support in CPU

[LXC](#) [2008] ([wikipedia](#))

OS virtualization

Linux only

[Docker](#) [2013] ([wikipedia](#))

Application containers

Linux and Microsoft Windows

I will spend quite a bit of time elaborating on each product, its design and how they differ... **verbally in class**. More detail will be offered in additional lectures as we concentrate on specific products.

Things not covered: [UML](#), [Wine](#), [Win4Lin](#), [QEMU](#), [Bochs](#)